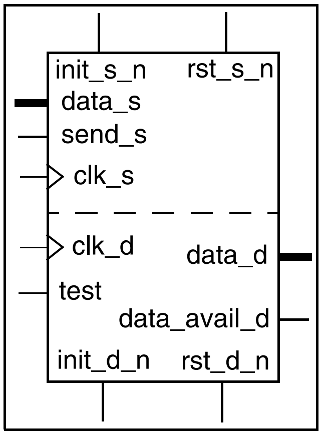
# DW_data_qsync_lh

## Quasi-Synchronous Data Interface for L-to-H Frequency Clocks

Version, STAR and Download Information: IP Directory

## Features and Benefits

- Fully tested clock domain crossing
- Fully parametrized
- Parameterized output registration: either combinatorial or registered outputs
- Applications: data bus controllers, bus-based communication circuits, any interface sending parallel data between two clock domains



## Description

This synchronizer passes data values from the low frequency domain to the high frequency domain. The two domains are synchronous with respect to each other with the low frequency clock expected to be a derivative of the high frequency clock. Therefore, the derived clock (slow clock) could contain jitter and skew uncertainties with respect to the originating clock source (the fast clock). The *clk_ratio* parameter integer value is determined by the high frequency divided by low frequency (or the slower clock's period divided by the faster clock's period). Full feedback handshake is not used. Note that when the *clk_ratio* is 2, the *tst_mode* parameter setting is ignored since a negative edge-triggered flip-flop is implemented between the two clock boundaries.

**Table 1-1 Pin Description**

| Pin Name | Width | Direction | Function |
|---|---|---|---|
| clk_s | 1 | Input | Source Domain clock source |
| rst_s_n | 1 | Input | Source Domain asynchronous reset (active low) |
| init_s_n | 1 | Input | Source Domain synchronous reset (active low) |
| send_s | 1 | Input | Source Domain send request input |
| data_s | *width* | Input | Source Domain send data input |
| clk_d | 1 | Input | Destination clock source |
| rst_d_n | 1 | Input | Destination asynchronous reset (active low) |
| init_d_n | 1 | Input | Destination synchronous reset (active low) |
| data_avail_d | 1 | Output | Destination data update output |
| data_d | *width* | Output | Destination data vector |
| test | 1 | Input | Scan test mode select |

### Table 1-2 Parameter Description

| Parameter | Values | Description |
|---|---|---|
| width | 1 to 1024<br>Default: 8 | Vector width of input `data_s` and output `data_d` |
| clk_ratio | 2 to 1024<br>Default: 2 | Integer value of the high speed clock period divided by the low speed clock period |
| reg_data_s | 0 or 1<br>Default: 1 | ■ 0 = Input data is not registered<br>■ 1 = Input data is registered |
| reg_data_d | 0 or 1<br>Default: 1 | ■ 0 = Output data is not registered<br>■ 1 = Output data is registered |
| tst_mode | 0 to 2<br>Default: 0 | Test mode<br>■ 0 = No latch is inserted for scan testing<br>■ 1 = Insert negative-edge capturing register on `data_s` input vector when `test` input is asserted<br>■ 2 = Reserved |

### Table 1-3 Synthesis Implementations

| Implementation Name | Function | License Feature Required |
|---|---|---|
| rtl | Synthesis model | DesignWare |

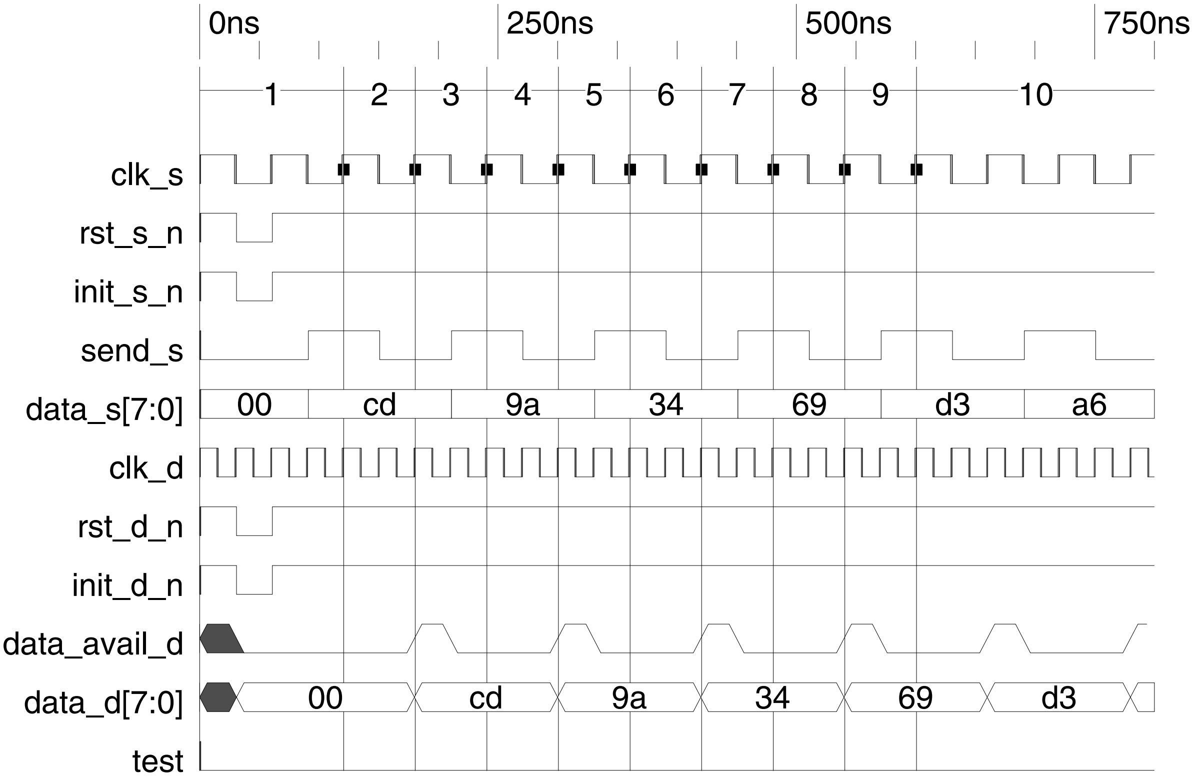
### Table 1-4 Simulation Models

| Model | Function |
|---|---|
| DW03.DW_DATA_QSYNC_LH_CFG_SIM | Design unit name for VHDL simulation |
| dw/dw03/src/DW_data_qsync_lh_sim.vhd | VHDL simulation model source code (modeling RTL)—no missampling |
| dw/sim_ver/DW_data_qsync_lh.v | Verilog simulation model source code |

## Timing Diagrams

The following diagram illustrates data transfer between source and destination. Data transferred by the source domain is indicated by the send_s pulse. Data on the bus not transferred is ignored.

**Figure 1-1 Timing Diagram**



## Related Topics

- Memory – Registers Overview
- DesignWare Building Block IP User Guide

## HDL Usage Through Component Instantiation - VHDL

```
library IEEE,DWARE;
use IEEE.std_logic_1164.all;
use DWARE.DW_Foundation_comp.all;

entity DW_data_qsync_lh_inst is
      generic (
        inst_width : NATURAL := 8;
        inst_clk_ratio : NATURAL := 2;
        inst_reg_data_s : NATURAL := 1;
        inst_reg_data_d : NATURAL := 1;
        inst_tst_mode : NATURAL := 0
        );
      port (
        inst_clk_s : in std_logic;
        inst_rst_s_n : in std_logic;
        inst_init_s_n : in std_logic;
        inst_send_s : in std_logic;
        inst_data_s : in std_logic_vector(inst_width-1 downto 0);
        inst_clk_d : in std_logic;
        inst_rst_d_n : in std_logic;
        inst_init_d_n : in std_logic;
        data_avail_d_inst : out std_logic;
        data_d_inst : out std_logic_vector(inst_width-1 downto 0);
        inst_test : in std_logic
        );
    end DW_data_qsync_lh_inst;


architecture inst of DW_data_qsync_lh_inst is

begin

    -- Instance of DW_data_qsync_lh
    U1 : DW_data_qsync_lh
    generic map ( width => inst_width,
                      clk_ratio => inst_clk_ratio,
                     reg_data_s => inst_reg_data_s,
                     reg_data_d => inst_reg_data_d,
                     tst_mode => inst_tst_mode )
    port map ( clk_s => inst_clk_s,
                   rst_s_n => inst_rst_s_n,
                   init_s_n => inst_init_s_n,
                   send_s => inst_send_s,
                   data_s => inst_data_s,
                   clk_d => inst_clk_d,
                   rst_d_n => inst_rst_d_n,
                   init_d_n => inst_init_d_n,
                   data_avail_d => data_avail_d_inst,
                   data_d => data_d_inst,
                   test => inst_test );


end inst;
```

## HDL Usage Through Component Instantiation - Verilog

```
module DW_data_qsync_lh_inst( inst_clk_s,
                              inst_rst_s_n,
                              inst_init_s_n,
                              inst_send_s,
                              inst_data_s,

                        inst_clk_d,
                              inst_rst_d_n,
                              inst_init_d_n,
                              data_avail_d_inst,
                              data_d_inst,

                        inst_test );

parameter width      = 8;
parameter clk_ratio  = 2;
parameter reg_data_s = 1;
parameter reg_data_d = 1;
parameter tst_mode   = 0;


input               inst_clk_s;
input               inst_rst_s_n;
input               inst_init_s_n;
input               inst_send_s;
input [width-1 : 0] inst_data_s;
input               inst_clk_d;
input               inst_rst_d_n;
input               inst_init_d_n;

output               data_avail_d_inst;
output [width-1 : 0] data_d_inst;
input                inst_test;

    // Instance of DW_data_qsync_lh
    DW_data_qsync_lh #(width,
                       clk_ratio,
                       reg_data_s,
                       reg_data_d,
                       tst_mode)
      U1 ( .clk_s(inst_clk_s),
               .rst_s_n(inst_rst_s_n),
               .init_s_n(inst_init_s_n),
               .send_s(inst_send_s),
               .data_s(inst_data_s),
               .clk_d(inst_clk_d),
               .rst_d_n(inst_rst_d_n),
               .init_d_n(inst_init_d_n),
               .data_avail_d(data_avail_d_inst),
               .data_d(data_d_inst),
               .test(inst_test) );

endmodule
```

# Copyright Notice and Proprietary Information

## Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

## Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

## Trademarks

Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at https://www.synopsys.com/company/legal/trademarks-brands.html.

All other product or company names may be trademarks of their respective owners.

## Free and Open-Source Software Licensing Notices

If applicable, Free and Open-Source Software (FOSS) licensing notices are available in the product installation.

## Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

Synopsys, Inc.
www.synopsys.com